AMENDMENTS TO THE CLAIMS

Claims 1-62 (Canceled).

63. (Previously Presented) A method of passing Internet Protocol (IP) data packets through a network, said method comprising:

constructing a chunk as a substantially fixed quantity of data with a payload that is sized to fit more than one of said data packets;

filling said payload of said chunk with a portion of at least one data packet;

including a framing symbol in each said chunk

converting said chunk from electrical information into optical information; and

passing said chunk through an optical switch fabric.

64. (Previously Presented) The method of claim 63 further comprising inserting said framing symbol adjacent to the trailing end of said chunk.

65. (Previously Presented) The method of claim 63 wherein said passing comprises using said framing symbol to determine uniquely within a stream of bits the beginning and the trailing end of said chunk.

66. (Previously Presented) The method of claim 63 wherein said framing symbol has a length of 16 bytes.

67. (Previously Presented) The method of claim 66 wherein said framing symbol includes two bits of said framing symbol intermixed in each of 56 contiguous bytes of said chunk immediately followed by two contiguous bytes of said framing symbol.

68. (Previously Presented) The method of claim 63 further comprising:

formatting said chunk to include forward error correction (FEC) coding.

69. (Previously Presented) The method of claim 68 wherein said formatting includes cyclical redundancy check (CRC) coding in each chunk.

70. (Previously Presented) The method of claim 69 further comprising using said FEC encoded in each said chunk to detect and correct errors in said chunk.

71. (Previously Presented) The method of claim 70 further comprising using said CRC encoded in each said chunk to determine that the entire said chunk has a proper CRC value.

72. (Previously Presented) The method of claim 63 further comprising:

formatting said chunk to include a scrambler seed, and wherein said formatting comprises using said scrambler seed in said chunk to balance zeroes and ones and to minimize run lengths of zeroes and ones by scrambling bits across said chunk.

73. (Previously Presented) The method of claim 63 further comprising:

formatting said chunk to include a "Break Bytes" field and a "Make Bytes" field, said fields configured to precondition an optical receiver prior to the arrival of said chunk.

74. (Previously Presented) The method of claim 73 wherein said "Break Bytes" field and said "Make Bytes" field are programmable in length.

75. (Previously Presented) The method of claim 73 wherein said passing comprises using said "Break Bytes" field and said "Make Bytes" field to precondition an optical receiver prior to the arrival of a chunk.

76. (Previously Presented) The method of claim 75 wherein said "Break Bytes" field maintains a 50 percent density of ones and zeros for a laser beam.

77. (Previously Presented) The method of claim 75 wherein said "Make Bytes" field reestablishes a decision threshold level of a limiting amplifier within a burst mode optical receiver.

78. (Previously Presented) The method of claim 63 further comprising:

formatting said chunk to include adding a chunk header.

79. (Previously Presented) The method of claim 78 wherein said chunk header includes identification of chunk type.

80. (Previously Presented) The method of claim 78 wherein said chunk header includes a header parity.

81. (Previously Presented) The method of claim 78 wherein said chunk header includes an indication that said chunk is a master chunk.

82. (Previously Presented) The method of claim 78 wherein said chunk header includes a sequence number.

83. (Previously Presented) The method of claim 82 further comprising:

performing error detection and correction using said sequence number in said chunk header for alarming and for alerting that a chunk potentially was corrupted.

84. (Previously Presented) The method of claim 83 wherein a re-initialize bit is used to enable reinitialization of said sequence number, such that said alarming is avoided.

85. (Previously Presented) The method of claim 63 wherein said chunk has a length of approximately 400 bytes.

86. (Previously Presented) The method of claim 63 wherein said chunk contains multiple data packets.

87. (Previously Presented) The method of claim 63 wherein said sized chunk contains a segment of a data packet, said data packet having a length greater than the size of said chunk.

88. (Previously Presented) An Internet Protocol (IP) packet router system, said system comprising:

at least one chunk having a payload comprising a plurality of data packets and a framing symbol; and

an IP packet router, including:

an optical switch fabric through which said chunk passes; and;

a first electrical switch stage at an input side of said optical switch fabric and a second electrical switch stage at an output side of said switch fabric.

89. (Previously Presented) The IP packet router system of claim 88 wherein said first electrical switch stage is operable to construct said chunk, and said second electrical switch stage is operable to strip said data packets from said chunk.

90. (Previously Presented) The IP packet router system of claim 88 wherein said framing symbol has a length of 16 bytes, and wherein said framing symbol is located adjacent the trailing end of each said chunk.

91. (Previously Presented) The IP packet router system of claim 90 wherein said framing symbol includes two bits of said framing symbol intermixed in each of 56 contiguous bytes of said chunk immediately followed by two contiguous bytes of said framing symbol.

92. (Previously Presented) The IP packet router system of claim 91 wherein each said chunk is formatted to include a chunk cyclical redundancy check (CRC) field.

93. (Previously Presented) The IP packet router system of claim 92 wherein said chunk CRC field is located adjacent and preceding said two contiguous bytes of said framing symbol.

94. (Previously Presented) The IP packet router system of claim 88 wherein each said chunk is further formatted to include forward error correction (FEC) coding.

95. (Previously Presented) The IP packet router system of claim 88 wherein said FEC coding is located adjacent to and following said framing symbol.

96. (Previously Presented) The IP packet router system of claim 88 wherein each said chunk is formatted to include a preamble, said preamble containing information configured to allow alignment of router clock and data recovery circuitry.

97. (Previously Presented) The IP packet router system of claim 96 wherein each said chunk is formatted to include a "Break Bytes" field and a "Make Bytes" field, said fields configured to precondition an optical receiver prior to the arrival of a chunk.

98. (Previously Presented) The IP packet router system of claim 97 wherein said "Break Bytes" field is located ahead of said "Make Bytes" field in a chunk.

99. (Previously Presented) The IP packet router system of claim 98 wherein said "Make Bytes" field is located ahead of said preamble.

100. (Previously Presented) The IP packet router system of claim 88 wherein each said chunk is formatted to include a scrambler seed.

101. (Previously Presented) The IP packet router system of claim 88 wherein each said chunk is formatted to include a chunk header.

102. (Previously Presented) The IP packet router system of claim 101 wherein said chunk header includes identification of chunk type.

103. (Previously Presented) The IP packet router system of claim 101 wherein said optical switch fabric is partitioned into a plurality of working subplanes.

104. (Previously Presented) The IP packet router system of claim 103 wherein said chunk header includes identification of a specific routing subplane through said switch fabric.

105. (Previously Presented) The IP packet router system of claim 101 wherein said chunk header includes a header parity.

106. (Previously Presented) The IP packet router system of claim 101 wherein said chunk header includes identification of an input of said optical switch fabric and an output of said optical switch fabric for said chunk.

107. (Previously Presented) The IP packet router system of claim 101 wherein said chunk header includes a master chunk bit.

108. (Previously Presented) The IP packet router system of claim 101 wherein said chunk header includes a sequence number.

109. (Previously Presented) The IP packet router system of claim 88 wherein said payload of said at least one chunk further comprises at least one packet segment and an associated packet header.

110. (Previously Presented) The IP packet router system of claim 88 wherein said at least one packet segment is selected from the group consisting of portions of packets that are larger than said chunk payload, portions of packets that are equal in size to said chunk payload, and portions of packets that are smaller in size than said chunk payload.

111. (Previously Presented) The IP packet router system of claim 88 wherein said switch fabric comprises a plurality of optical switch planes.

112. (Previously Presented) A method of information flow through an IP packet network system, said method comprising:

encapsulating input data packets from a plurality of source ports into substantially fixed sized chunks, wherein said input data packets are electrical signals;

formatting overhead information onto each of said chunks, said overhead including a framing symbol;

electrically switching said chunks to be sent to an appropriate optical switch plane;

converting said chunks into optical signals; and

directing said chunks through said appropriate optical switch plane toward a plurality of destination ports.

113. (Previously Presented) The method of claim 112 further comprising:

converting said directed chunks into electrical signals;

electrically switching said chunks from said appropriate optical switch plane;

performing error detection and error correction on said chunk;

removing said overhead information from said chunks; and

reassembling said input data packets out of said chunks.

114. (Previously Presented) The method of claim 112 wherein all information flows through said switch plane in said substantially fixed sized chunks.

115. (Previously Presented) The method of claim 112 further comprising:

formatting said chunks to include adding a chunk header.

116. (Previously Presented) The method of claim 115 wherein said appropriate switch plane is one of a plurality of subplanes comprising a partitioned switch fabric.

117. (Previously Presented) The method of claim 116 wherein said chunk header includes identification of a specific routing subplane through said switch fabric.

118. (Previously Presented) The method of claim 117 wherein said directing comprises using said identification in said chunk header of a specific routing subplane to route said chunks through said switch fabric.

119. (Previously Presented) The method of claim 115 wherein said chunk header includes identification of an input of said appropriate optical switch plane and an output of said appropriate optical switch plane for said chunk.

120. (Previously Presented) The method of claim 119 wherein said directing comprises using said identification in said chunk header of said input and said output to route said chunks through said optical switch plane.

121. (Previously Presented) The method of claim 119 further comprising:

performing error detection and correction using said identification in said chunk header of said input and said output to verify the route of said chunks from said input and said output.

122. (Previously Presented) The method of claim 115 wherein said chunk header includes identification of chunk type.

123. (Previously Presented) The method of claim 122 wherein said directing comprises using said identification of chunk type in said chunk header to enable guaranteed bandwidth chunks to pass ahead of best effort chunks through said switch plane.

124. (Previously Presented) The method of claim 112 wherein said optical switch plane is part of an optical switch fabric.

125. (New) The method of claim 112 wherein said electrically switching comprises using said framing symbol in each said chunk to determine uniquely within a stream of bits a beginning and a trailing end of each said chunk.